基于拟态计算的大数据高效能平台设计方法 *

李 斌[1]，周清雷[2]，斯雪明[1]，聂 凯[1]

(1. 信息工程大学 数学工程与先进计算国家重点实验室，郑州 450001; 2. 郑州大学 信息工程学院，郑州 450001)

摘 要：针对当前大数据应用主要以通用处理器为计算核心，且系统结构单一、能效比低，无法充分满足大数据的计算需求。基于拟态计算模型，提出了一种大数据高效能平台的设计方法。该方法以算粒为基本研究对象，深入剖析大数据应用算法的特征，合理划分各计算子任务；其次，构造体系结构匹配矩阵，并将子任务分配到合理的处理部件上；最后，利用动态电压/频率调节技术和数据布局算法，实现非关键任务的电压控制，并优化关键任务的结构布局。实验结果表明，拟态计算能深度融合各异构计算部件，建立具有灵活、可拓展的体系结构，充分发挥系统整体执行效率，降低功耗，提高能效比。

关键词：大数据；拟态计算；算粒；匹配矩阵；能效比

中图分类号：TP doi: 10.3969/j.issn.1001-3695.2018.02.0084

Design method of big data high-efficiency platform based on mimic computing

Li Bin[1], Zhou Qinglei[2], Si Xueming[1], Nie Kai[1]

(1. *State Key Laboratory of Mathematical Engineering & Advanced Computing*, *Information Engineering University*, *Zhengzhou* 450001, *China*; 2. *School of Information Engineering*, *Zhengzhou University*, *Zhengzhou* 450001, *China*)

Abstract: In view of the current big data applications mainly use the general processor as the computing core, and the system structure is simple, energy efficiency ratio is low, can't fully meet the big data computing needs. In this paper, based on mimic computing model, a design method of big data high-efficiency platform is put forward. This method took computing grain as the basic research object, deeply analyzed the features of big data application algorithms, and reasonably divided the computational subtasks. Secondly, an architecture matching matrix was constructed and the subtasks were assigned to the right processing units. Finally, dynamic voltage/frequency scaling technology and data layout algorithm were used to control the voltage of non-critical tasks and optimize the structure layout of critical tasks. The experimental results show that the mimic computing can integrate the heterogeneous computing components in depth, establish a flexible and scalable architecture, give full play to the overall efficiency of the system, reduce the power consumption and improve the energy efficiency ratio.

Key words: big data; mimic computing; computing grain; matching matrix; energy efficiency ratio

0 引言

近年，随着大数据技术的飞速发展，“数据为王”的时代已经到来。大数据中蕴藏的宝贵价值，在社交、金融、医疗、电信等领域引起了人们高度的重视。但是，海量、模态多样的非结构化数据，使得大数据环境的构建颇为复杂，这要求在计算架构和大规模数据处理机制上实现范式转变。同时，大数据的传输、存储和分析处理都将消耗大量的能源，研究创新的节能计算技术，也是亟待解决的问题[1]。

拟态计算以实现高效能和高性能计算为目的，具有按需分配资源、结构可变、灵活计算的特点。拟态计算通过识别应用的需求、应用的变化，同时感知系统中可以利用的处理资源，依据尽可能高效的原则，构建出适合于应用需求的处理结构，并且该结构随着应用的变化，如：计算进展阶段、处理负荷等的变化，而进行结构的主动变更，达到“应用决定结构，结构决定效能”的目的。拟态计算能充分利用程序和计算部件的异构性，各尽潜能，合理分治，协同计算一个应用任务，兼顾性能和灵活性。本文基于拟态计算的思想，提出了一种大数据高效能平台设计方法，通过分析大数据应用的算粒特征，依据尽可能高效的原则，合理利用系统中的处理资源，构建出适合当

收稿日期：2018-02-07；**修回日期：**2018-03-21 **基金项目：**国家重点研发计划资助项目（2016YFB0800100，2016YFB0800101）；国家自然科学基金资助项目（61250007）；国家“863”计划资助项目（2009AA012201）

作者简介：李斌（1986-），男，河南郑州人，博士研究生，主要研究方向为高性能计算和信息安全（cctvlibin@163.com）；周清雷（1962-），男，河南新乡人，教授，博导，主要研究方向为信息安全、自动机理论及计算复杂性理论；斯雪明（1966-），男，福建福州人，副教授，主要研究方向为密码学、网络安全和高性能计算；聂凯（1987-），男，河南镇平人，博士研究生，主要研究方向为信息安全和形式化方法.

前大数据应用的处理结构，并利用动态能效的优化，在保证计算性能的同时降低系统功耗。进而，为创新大数据计算模拟环境，提供理论基础，降低大数据应用开发和利用的门槛。

1　相关研究

大数据是一个新生事物，目前在国内外并没有出现较为成熟的大数据计算平台。文献[2]通过分析大数据在内存和网络方面的结构特征，提出了使用 FPGA 构建集群，解决大数据所面临的问题。文献[3]通过调研现有大数据处理平台，给出了基于应用特性的硬件系统配置方案，相比较于 CPU 和 GPU，在大数据处理中 FPGA 具有并行性、流水线和高性能的特点。文献[4]通过分析现有大数据处理架构所存在的问题，给出了混合异构平台在加速比、扩展性和灵活性等方面的优势。文献[5]通过构建异构平台 CPU+FPGA 在数据挖掘和深度学习算法方面，取得了较好的加速比和能效比，并降低了系统功耗。文献[6]提出了面向大数据应用的异构多核可重构平台，通过可重构器件和高性能通用处理器总线互连，利用多计算资源并行执行的调度算法，使所有计算资源(CPU+FPGA)共同并行地执行任务，提高了大数据计算效率。文献[7]提出了分布式流体系结构 DSA（distributed stream architecture）及其编程模型与资源管理，并在 CPU 和 GPU 异构系统上实现了原型系统，提高了计算性能。文献[8]分析了大数据负载的体系结构特征，为大数据平台的处理器设计、算法优化具有指导意义。文献[9]通过分析大数据平台所需要的扩展性、一体化和多样性需求，采用硬件定制化的设计和混合型软件架构支持多种大数据应用类型。

由上，当前对大数据体系结构的研究主要集中在混合异构平台。但是上述方案并未考虑在异构系统中，大数据应用的算法特征及计算任务与体系结构匹配的问题。另外，随着计算系统越来越庞大，平台的维护和运营的代价也越来越高，处理任务的能效比成为了用户关心的重要因素。因此，大数据的研究应用，急需一个创新的平台来支撑全生命周期内跨领域、异构大数据的管理、分析和处理等需求[9]。

2　基于拟态计算的大数据高效能平台设计方法

2.1　拟态计算模型

包含软件和硬件变体的多维重构函数化体系结构称为拟态架构，它能根据动态参数选择生成多种功能等价的可计算实体，实现拟态变换。对于一个确定的可计算问题，在拟态架构中可以由多种功能等价、计算效能不同的硬件变体和软件变体来实现，动态地选择与使用这些变体，计算效能可以达到最优化，即为拟态计算[11]。

拟态计算可以根据大数据应用任务的特征、QoS 的要求、可用的资源，构建出最优的应用结构，实现应用到方案的最优映射。拟态计算模型，如图 1 所示，可抽象为七元组（*APP*,*MA*,*OA*,*SE*,*EF*,*KB*,*DS*）其中：

APP 为应用的集合，它包含应用的名称、类型、功能、PMC(Processing-Memory-Communication)需求、负荷、服务质量要求等属性；

MA 为元结构的资源集合，它描述资源的 ID，以及相应的 PMC 属性；

OA 为应用目标结构集合，集合中的元素是对应于应用的高效结构，它是元结构 *MA* 的一个子集；

SE 为系统状态集合，它动态的反映每个应用和应用结构的状态，包括 PMC 需求、应用负荷、功耗、资源利用率等；

EF 是评价函数，可以用来评价每次重构出来的应用结构在 QoS 方面的表现，例如性能、效能、安全性等；

KB 为知识库，库中包含推理知识、推理规则、方案策略等内容。知识库具有自学习功能，在系统运行过程中通过自学习不断进行更新；

DS 为认知决策函数，用于决策应用的目标结构。*DS* 以 *EF* 趋向于最高为原则，综合利用 *APP*、*MA*、*EF*、*KB* 等要素，根据 *SE* 动态的决策出应用的高效能结构 *OA*。

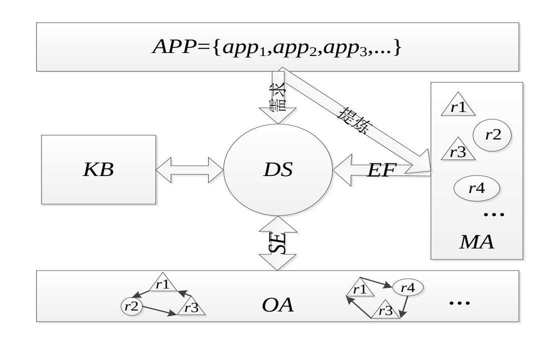


图 1　拟态计算模型

由拟态计算的定义和模型可知，它采用了“研究和建立最合适的计算模型、使用和构建最合适的处理部件、设计和匹配最合适的体系结构、追求和逼近最理想的综合效能”的技术思想，可以根据大数据应用的特征、QoS 的要求、系统的状态，利用基于知识库的决策函数，对应用的属性、服务质量、负荷变化以及系统构件资源进行感知，并选取合适的计算部件，搭建异构多核的高效能大数据平台系统。

拟态计算模型揭示了“刚性不变的体系结构支持差异巨大的应用是使计算效能低下的根本原因”，其本质是以拟态变换实现高效能的计算。对于不同需求的应用，其复杂度、处理负荷、效能需求、处理时机、应用特点等因素的不同可导致对应的高效软件、硬件及体系结构解算方案的不同。而传统大数据平台在处理架构上基本是确定的，严重束缚了应用运行时效能的改善。拟态计算通过对应用和资源的静态或动态感知，在主动认知的基础上，采取在合理的预先时机选取或重构出合适的资源及结构，动态地为应用提供最合适的体系结构方案和执行方案，力求不断逼近最优效能的应用计算需求，从而达到高效能的目标。

此外，异构多核计算能够在保证系统通用性的前提下提供更具效能比的计算平台，也是未来大数据系统结构发展的方向。鉴于此，拟态计算模型立足于选择多种灵活的计算资源和存储

资源接入方式，尽可能多地将各类计算资源和存储资源方便地纳入体系结构并构成一个有机整体，如已有成熟的 CPU、GPU、FPGA 等处理器，并利用资源的聚合来处理应用中蕴涵的不同计算组合需求，以达到面向应用高效能计算。显然，拟态计算可为大数据应用平台的搭建提供强有力的支持。

为有效将拟态计算模型应用于大数据领域，构建高效能异构计算平台，需要首先对大数据应用进行算粒特征分析，由 PMC 属性和执行频率等参数合理的划分各运算段。然后，从系统可用元结构资源集合中，依据能效比最优，建立体系结构匹配矩阵，并选取各运算段适合的目标结构。再结合已有的经验和知识，根据系统负载的变化，动态调节系统电压和频率，并优化数据布局，降低能耗。以下给出了具体的实现流程及方法。

2.2　大数据应用算粒的特征分析

2.2.1 算粒的定义

为了实现大数据应用灵活、可拓展和高效能计算，拟态计算需要对多种算法进行算粒特征分析，研究大数据算粒彼此间的关系和共性，为系统决策提供依据。

计算粒子，简称算粒，是对某一计算处理序列或算法结构的粗粒度、层次化表示方法，是传统计算指令到算法粒化结构的对应。算粒通过不同计算尺度，对计算、处理过程进行分解，形成数据和功能层面上的计算片段集合，根据相似性和功能近似性，进行整合。算粒反映了大数据应用计算特征的一种模式，是完成任务算法结构的一种抽象和归纳[12]。它具有独立性、可变粒度、普适性和 PMC 属性。

例如，排序、FFT、矩阵乘法等基本算法流程，甚至更大计算尺度的处理过程，如图像处理等都可以认为是某种算粒。显然，也可以根据实际执行计算功能部件的单位处理能力，将上述算粒分解成更小的算粒，如 32 位进位保留加法器、移位乘法组合、逻辑组合函数、乘后加等。

2.2.2 算粒计算模型

算粒计算模型可表示为：$CG=(V,E,P,M,C,U)$，其中 $V=\{v_1,v_2,\ldots,v_n\}$，表示算粒的集合；$E=\{e_{ij}\mid e_{ij}=(v_i,v_j),\ v_i,v_j\in V,1\le i,j\le n\}$，称为有向边集，表示算粒之间数据依赖先后关系及串并行关系；$P=\{p_1,p_2,\ldots,p_n\}$，代表算粒的串行计算量；$M=\{m_1,m_2,\ldots,m_n\}$，代表算粒所需内存容量或硬件寄存器资源；$C=\{c_1,c_2,\ldots,c_n\}$，代表每条边 e_{ij} 上的通信量；$U=\{u_1,u_2,\ldots,u_n\}$ 表示计算部件，$u_i\in\{CPU,GPU,CELL,DSP,\ldots,1\le i\le n\}$。

对于算粒计算模型，其计算粒度的规模受解算目标的应用驱动，计算粒度越大，性能越好，计算量粒度越小，灵活性越高，因此算粒兼顾高效计算和灵活性。

2.2.3 算粒的分析方法

结合大数据应用特征及算粒模型，采用控制数据流图 CDFG(Control Data Flow Graph)描述程序计算任务之间算粒的划分、并行和依赖关系。CDFG 图类似于有向图，可以准确的描述程序中代码的执行顺序和数据的传递过程[13]。CDFG 图中最小的节点单元是基本块 *BB*（Basic Block）,每一个 *BB* 都可以被认为是一个小的 DFG 图。CDFG 图较 DFG 图包含的信息更广，更有利于从大数据应用整体，分析程序的执行行为。CDFG 图的定义如下[14]：

$$CDFG=(V,E)$$

$$V=\{BB_1,BB_2,\ldots,BB_n\}$$

$$E=\{e_1,e_2,\ldots,e_m\}$$

其中:*V* 是基本块 *BB* 的集合，且每一个 *BB* 是一个小的 DFG 图，包含基本操作和数据流向；*E* 是边的集合，每一条边 *e* 表示的是 *BB* 间的跳转过程。如图 2 所示，展示了 C/C++代码与 CDFG 图之间的关系。

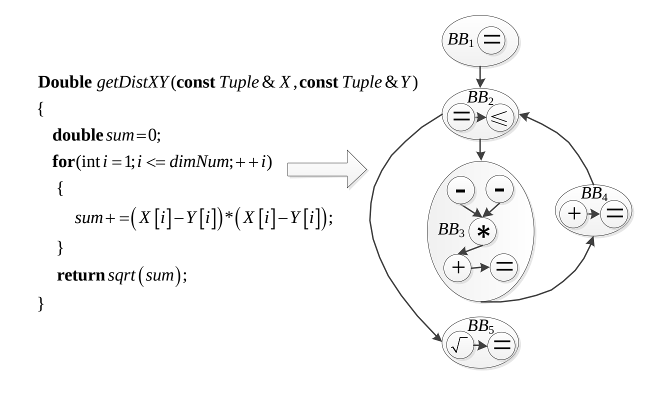


图 2 C/C++代码与其 CDFG 图

得到大数据应用的 CDFG 后，可以通过 peeling 算法[13]对其进行划分，形成子图。由于子图包含操作数和程序执行片段，具有 PMC 属性，所以符合算粒的模型描述。为了保证子图有效的划分，需要满足以下约束条件。

a)凸性。凸性是指 CDFG 子图和 DFG 子图必须是凸子图，即子图中不存在一个节点连接到子图外的节点上，然后再连接回该子图中的节点。这是为了保证生成算粒的完整性和结果的唯一性。

b)连通性。连通性是指 CDFG 图和 DFG 图是一个连通图。如果子图不是连通图，就无法确定每一个操作的执行顺序，也不能保证数据正确的传递，将造成算粒划分的结果不唯一。

根据不同大数据应用间划分的 CDFG 子图，按照代码相似性和功能近似性，进行整合，形成基础算粒集。通过对基础算粒集的动态粗粒度重构，可以灵活地拓展形成相应的程序，适应多种大数据应用的需求。

进一步，通过对算粒集的分析，找到执行频率高的 *BB* 节点和频率低的 *BB* 节点所在的算粒。并将执行频率高的 *BB* 节点所在的算粒，确定为高阶运算段，即关键路径，并做相应的优化处理，如在可重构硬件上，通过流水线的方式实现。同时，还可为专用加速指令等设计提供有效的指导，提高处理器的性能。

2.3　大数据计算任务与体系结构匹配矩阵

对某一大数据应用 *BD.APP*＝{*Name*, *Type*, *Function*, *PmcR*, *Load*, *Qos*}，将其按过程函数中粒度算粒划分为 *m* 个计算子任务，记为 $task=\{f_1,f_2,f_3,\ldots,f_m\}$，每个 $f_i(1\le i\le m)$ 为可完全独立

执行的代码片段，由单个或多个基本算粒构成，其基本操作为顺序执行或并发执行。同时，对于拟态计算元结构资源集合 $BD.MA=\{r_1,r_2,r_3,...,r_n\}$，其中每个 r_j 由计算资源 p_x、存储资源 m_y、互连结构 c_z 三元组构成，记为 $r_j=\{p_x,m_y,c_z\},(1\le j\le n)$，且 $p_x\in\{CPU,GPU,CELL,DSP,FPGA,RMS,ASIC,...\}$，$m_y\in\{SRAM,DRAM,NVRAM,SCM,SSD,SATA,...\}$，$c_z\in\{L,C,R\}$，元素 *L* 代表线直连，*C* 为开关互连，*R* 为路由连接。这样，将性能各异的处理器、存储设备、互连结构通过高速网络连成并行环境，充分利用计算子任务和结构的异构性，协同完成一个大数据应用任务，使得系统能效最高，如图 3 所示。

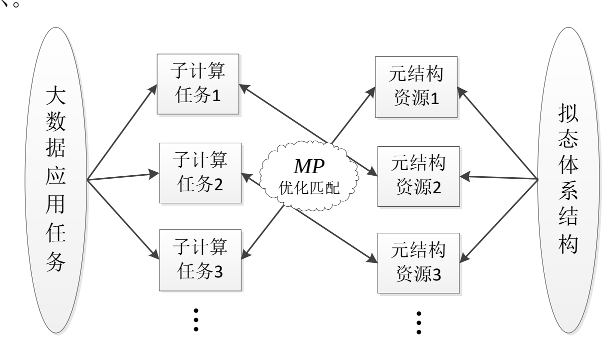


图 3 大数据任务和体系结构匹配

那么，*m* 个计算子任务分配到由 *n* 个异构处理器组成的拟态计算系统，有多种分配方案。不同的分配方案，系统性能也各不相同，甚至差距很大。因此，在拟态计算系统中，需要将计算任务模式和体系结构相互匹配。这里，通过计算任务和体系结构匹配矩阵进行描述[15]，记 $MP=(p_{ij})_{m\times n}$，其中 $1\le i\le m,1\le j\le n$，如图 4 所示，p_{ij} 是计算子任务 f_i 在元结构处理器 r_j 上的相对性能表现。

$$\begin{array}{c} \\ f_1 \\ f_2 \\ \cdots \\ f_m \end{array}\begin{array}{c} \begin{matrix} r_1 & r_2 & \cdots & r_n \end{matrix} \\ \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ p_{m1} & p_{m2} & \cdots & p_{mn} \end{bmatrix} \end{array}$$

图 4 计算任务和体系结构匹配矩阵

进一步，针对大数据计算任务模式的多样性，例如分布式计算、内存迭代计算、流式数据处理、实时数据查询等，需要从各个角度对 p_{ij} 进行综合评估，从而完成系统的深度融合，充分发挥拟态系统的整体执行效率。为此，可以结合先验知识，通过执行速度、功耗、延迟和存储 I/O 等方面来对 p_{ij} 进行评价，并从中选取合适的体系结构。

假设用工作负载 *W*，表示求解计算任务 *task* 的工作量（与输入操作数相关），记 $EF(MP)=EER(W)$，表示评估当前拟态系统的整体能效。显然，能效比越高，系统的表现越好。对于每个计算子任务 f_i 的工作量 w_i，有 $EER(w_i)=\dfrac{perf(w_i)}{power(w_i)}$ 且 $W=\sum_{i=1}^{m}w_i$，其中 *perf* 表示性能，包括速度、延迟、I/O 等；*power* 表示功耗。另外，如果子任务 f_i 和 f_j ($i\ne j$)需要通信，且都在同一处理器上运行，则通信代价 0；否则需要在能效比中额外计算通信代价 *EER*(*comm*)。这样，通过评估每个子任务 f_i 在处理器 r_j 的能效表现 p_{ij} 和通信代价，进而得到整体 $EF(MP)$。然后采用模拟退火算法，设计约束条件 *EER*(*W*)最高，从 *MP* 中遍历当前大数据应用的结构 *BD.MA'*，并选取最匹配的体系结构 *BD.OA*。具体描述算法如下算法 1 所示，其中 *T* 为初始温度，$T_{\min}$ 为终止低温，Δt 为降温系数，$EER(W)_{opt}$ 为最优能效比。

算法 1. 大数据体系结构的匹配优化

Input: $BD.APP=\{Name, Type, Function, PmcR, Load, Qos\}$//大数据应用

Output: $BD.OA=\{oa_1,oa_2,oa_3,...,oa_m\}$ //目标元结构集合

1. initial $task=\{f_1,f_2,f_3,...,f_m\}$ //初始化子任务集合

2. initial $BD.MA=\{r_1,r_2,r_3,...,r_n\}$ //初始化可用元结构集合

3. 由 *task* 和 *BD.MA* 建立匹配矩阵 $MP=(p_{ij})_{m\times n}$

4. $W=0$ //初始化工作负载

5. for *i* = 1 to *m* do//评估每个子任务 f_i 在处理器 r_j 上的表现 p_{ij}

6. 　由 f_i 的输入操作数，计算工作量 w_i

7. 　$W=W+w_i$ //累加子任务的工作负载

8. 　for *j* = 1 to *n* do

9. 　　评估工作量 w_i 在 r_j 上的能效表现 p_{ij}，有 $EER(w_i)=\dfrac{perf(w_i)}{power(w_i)}$

10. 　end for

11. end for

12. while ($T>T_{\min}$) do//采用拟退火算法，找到最优能效比及对应的结构

13. 　由模拟退火算法，从 *MP* 每行中随机选取一个处理器 r_j，有 $BD.MA'=\{r_1',r_2',r_3',...,r_m'\}$，构成完整的应用

14. 　$EER(W)=EER(BD.MA')$ //计算当前结构对应的整体能效比

15. 　计算元结构互连通信代价 $comm(r_s,r_t)$，如果 $r_s=r_t$，则通信代价为 0；否则 $EER(W)+=EER(comm(r_s,r_t))$

16. 　if ($EER_{opt}(W)\le EER(W)$) do//最优能效比小于当前结构的能效比

17. 　　$BD.OA=Opt(BD.MA')$ //更新目标元结构集合

18. 　　$EER_{opt}(W)=EER(W)$ //更新最优能效比

19. 　end if

20. 　$T=T\times\Delta t$ //降温

21. end while

当算法 1 结束时，有 $EF(MP)=EER_{opt}(W)$ 表现最优。显然，对于给定的应用，在能效最优的约束条件下，建立合适的拟态混合体系结构，可完成应用的高效实现。

2.4　大数据体系结构能效的动态优化

在大数据应用中存在着大量的数据存取，处理器需要频繁地访问内存，导致在程序执行的过程中存在着高频率的计算和存储的相位切换[16]。而当前处理器的架构都是面向计算设计的，与大数据应用需求不匹配，资源利用率低。据调查显示，处理器加上内存的功耗就占到整个数据中心功耗的30%以上[17]。

动态电压/频率调节 DVFS[18] (dynamic vo1tage/ frequency scaling)是一种广泛使用的动态功耗优化技术，它通过在一定范围内降低处理器的电压/频率，以减少其能量的消耗。因此，可以采用电压动态调节技术，划分出不同的电压频率区域，实现线程的细粒度电压控制，从而进一步降低功耗。

处理器的功耗分为静态功耗和动态功耗，静态功耗主要与电路的漏电等相关，而动态功耗主与电压、节点电容和频率等相关。所以提高 clock 频率，在提高处理器性能的同时，也会提高处理器的动态功耗。处理器的动态功耗有下式计算[19]：

$$P_{dyn} = \alpha C_L V^2{}_{DD} f$$

其中:α反映电路的信号翻转率，C_L 是电容负载，V_{DD} 是供电电压，f 为频率。

对于拟态系统，整体功耗为 $P_{sys} = P_{dny} + P_{stc} = (P_{PRCS} + P_T + P_{IO})_{dny} + P_{stc}$，其中 P_{PRCS} 表示处理器功耗，P_T 表示通信功耗，P_{IO} 表示存取功耗。在不影响程序整体执行效率的情况下，可以对非关键任务划分电压频率区域，选择合适的时机，进行电压频率调节。而对于关键任务，利用数据布局算法，可以适当地增加缓存，降低 I/O 频率。假设已知初始目标元结构 *oa* 的状态，据此可在满足其他模块的需求下，优化调整 *oa* 的电压频率和结构布局，具体流程如算法 2 所示。

算法 2. 优化调整 *oa* 的电压频率和结构

Input: $BD.OA = \{oa_1, oa_2, oa_3, ..., oa_m\}$ //初始元结构集合

Output: $BD.OA' = \{oa_1', oa_2', oa_3', ..., oa_m'\}$ //优化调整后的元结构集合

1. for i = 1 to m do//对每个元结构进行调整优化
2. 　if (oa_i 上运行的为非关键任务)
3. 　　在满足时间的约束下，动态调节该处理单元的频率幅度，降低 P_{PRCS} 的功耗，得到优化后的元结构 oa_i'
4. 　else (oa_i 上运行的为关键任务)
5. 　　如果 oa_{i-1} 和 oa_i 或 oa_i 和 oa_{i+1} 需要通信，在满足 oa_{i-1} 和 oa_{i+1} 的 I/O 需求的情况下，优化 oa_i 的存储、互连结构等，降低 P_T 和 P_{IO} 的功耗，得到 oa_i'
6. 　end if
7. end for

由于大数据采用分布式计算，各个处理单元相对比较独立，且数据的访问存在高并发、随机性和离散性。因此，还可以通过其他数据布局算法[20]，例如：优化大规模网络存储系统的并行度、容错性、可靠性；降低数据密集型应用的网络传输；对 RAID 磁盘阵列的改进等，进一步优化大数据系统的能耗。

3　实验结果与分析

本文实验，通过 CPU、GPU 和 FPGA 搭建拟态异构计算平台，并由万兆网络互连，三种计算部件的信息如表 1 所示，CPU 操作系统为 Linux，FPGA 编程软件为 Vavido 2015.4。其中，FPGA 为可重构器件，集成有万兆网络接口、PCIE 接口和 DDR3 内存，可直接通过万兆网络或插在主机的 PICE 插槽中，与 CPU 通信。CPU 和 FPGA 内存容量均为 24GB，GPU 显存为 6GB，可满足大数据的存取需求。该大数据平台环境可根据用户配置，选择合适的计算资源建立数据连接，以异构多核实现高效能计算。

表 1 各计算部件配置信息

计算部件	名称	配置信息
CPU	IBM X3650 M3	1 颗 6 核 CPU；型号：X5650 2.66GHz；内存：24GB
GPU	Nvidia Tesla M2075	核心频率：1.15 GHz；核心数：448；显存：6GB
FPGA	Xilinx Virtex-6 LX550T	片内资源 SLICES：85920；内存：24GB

同时，选取高频交易为应用对象，高频交易日内交易量巨大，对市场数据的响应延时在微秒级，需要实时解析数据包内容并入库。其交易协议又分为明文数据包和密文数据包，密文数据包对计算能力要求更为苛刻。显然，常规 CPU 无法胜任高频交易的处理需求。实验在 TCP/IP 协议处理、数据检索和密码算法计算这三个方面，分别对比了各处理部件在网络实时响应能力、内存吞吐能力、计算能力及能效比的优劣势，进而分析高频交易各环节所适合的处理结构。并根据初步分析结果，搭建了拟态异构高频交易大数据平台，优化了系统整体性能，降低了系统功耗，验证了本文方法的有效性。

3.1　各异构部件性能分析

在万兆网络下，由 CPU 客户端发送大量 TCP/IP 请求包，CPU 和 FPGA 作为服务端进行响应，其中 FPGA 集成 TOE（TCP Offload Engine）IP 核，最多支持 128 个连接，客户端收到响应结果的延迟对比如表 2 所示（单位微秒）。

表 2 CPU 和 FPGA 网络响应延迟对比

发包量 / 计算部件	1	10^2	10^3	10^4	10^5
CPU	405	552	700	758	1002
FPGA	341	356	365	562	846

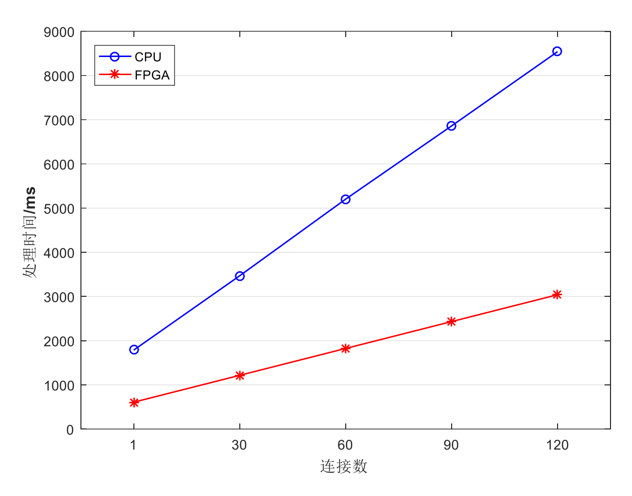


图 5 多连接下，CPU 和 FPGA 处理大量数据包时间对比

在多连接情况下，以 10 万 200 字节长度的数据包为一组，随着连接数的增加，CPU 和 FPGA 处理时间的变化，如图 5 所示。

从表 2 和图 5 中可以看出，在 TCP/IP 协议处理方面，FPGA 的延迟明显低于 CPU。这主要由于 FPGA 减少了传输层与网络层的数据拷贝，降低了协议复杂度，加快了网络传输速度。显然，使用 FPGA 作为网卡，在数据采集完后，可有效降低数据传输时延，为大数据的高速处理、高频交易和物联网提供了硬件支持。

在数据搜索方面，CPU 通过从硬盘读取大文件，并加载到内存，查找其中的关键字。FPGA 直接从万兆网络获取大文件，并传输到内存中，然后使用多模块并行查找关键字。两者的大文件处理速率及功耗，如表 3 所示。

表 3 CPU 和 FPGA 大文件搜索处理能力对比

计算部件	8G 大小的文件			64G 大小的文件		
	时间(s)	速率(MB/s)	功耗(W)	时间(s)	速率(MB/s)	功耗(W)
CPU	159.78	51.27	78	970.76	67.51	81
FPGA	36.44	224.8	56	194.16	337.54	57

从表 3 中可以看出，FPGA 在内存吞吐量及查询搜索上也有明显优势。这主要由于 FPGA 省去了操作系统的调度，并以板级总线直连内存，提高了内存读取效率。同时，多模块并行，也提高了数据搜索的速度。可见，对于海量结构化大数据存储检索系统，使用 FPGA 可有效提高数据吞吐量。

在密码算法计算方面，对于海量数据的管理，Hash 函数具有很好的压缩映射和等价索引功能，有效的降低了数据规模，例如对文件按 MD5 值来分表/分库，数据的查询去重等。下表对比了 CPU、GPU 和 FPGA 在 Hash 函数 MD5 下的计算性能。

表 4 CPU、GPU 和 FPGA 的 MD5 计算性能对比

计算部件	速度（个/s）	功耗（W）	能效比（个/s/W）
CPU	5045000	184	27418.5
GPU	1872250000	289	6478373.7
FPGA	1185500000	130	9119230.8

显然，GPU 和 FPGA 的计算能力要远高于 CPU，且 GPU 高于 FPGA，但由于 GPU 功耗较高，造成能效比较低。综合来看，FPGA 对密码算法的计算具有较好的表现，适用于大数据消冗，及大数据加密协议的处理，保障大数据的安全性。而在不考虑功耗的情况下，GPU 对大数据的加速处理更有优势。

由上，CPU 处理器善于通用事务处理、管理与调度，以及串行计算。GPU 由于并行计算能力强及算法实现相对 FPGA 简单特点，善于典型的复杂算法大规模并行运算，以及高精度浮点运算算法加速。FPGA 由于其大规模并行计算能力强，功耗低，算法实现相对复杂特点，善于数据采集、密码加解密等典型算法加速。所以，通过拟态计算技术，将 CPU 用于事务管理，融合 GPU 和 FPGA 加速技术，可以提高整个系统对大数据计算的效率。同时，各计算节点配以大容量高速存储，针对数据密集型应用，以降低数据传输及交互瓶颈。

3.2 拟态异构系统搭建

高频交易系统采用 FTD 协议，以明文和 3DES 加密数据包进行信息交互传输。通过对其应用算粒特征进行深入分析，发现数据包解析和 3DES 加解密耗费系统资源和时间较多，属于关键任务。进一步，数据包解析由报文头分类，内容解析，校验和纠错等组成，期望可多路并行，而 3DES 加解密主要有异或、S 盒、置换等操作组成，期望可高速并行。结合以上分析，可将这两个关键任务放置在 FPGA 上实现。以通过 CPU+FPGA 构成拟态异构系统，使用 FPGA 进行 TCP/IP 连接、数据包分类过滤、内容解析、校验和纠错、3DES 加解密，CPU 负责数据入库，其系统结构如图 6 所示。同时 CPU 端运行的是 Linux 系统，使用 tuned 系统调优工具，动态优化 CPU 频率、存储和网络延迟，达到优化系统、降低功耗的目的。在不增加 CPU 系统功耗的前提下，每秒交易处理性能约提升 6%。

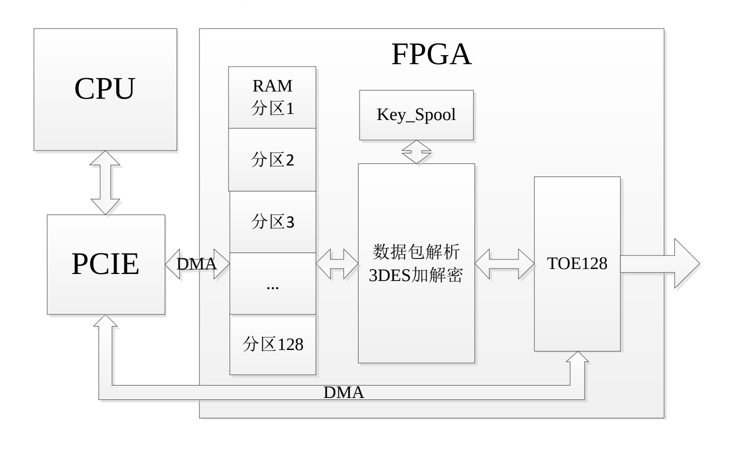


图 6 CPU+FPGA 拟态异构系统

FPGA 数据包解析和 3DES 加解密采用流水线结构实现，在满足核心模块处理需求的情况下，使用 RAM 进行分块存储，并优化各模块布局，缩短路径延迟。其中 Key_Spool 用来存储密钥，使用 1 个 RAM 实现，其它各模块说明及资源占用情况如下表所示：

表 5 FPGA 各模块资源占用情况

功能模块	说明	频率	占用资源（slices）
PCIE	CPU 与 FPGA 的数据通道，由系统底层驱动	250MHZ	6384
DMA	直接内存存取，实现数据的高速传输	250MHZ	2314
TOE128	TCP/IP 协议处理引擎，最多支持 128 个连接	156.25MHZ	4544
数据包解析	按报文头分类、提取内容，并完成校验和比对	156.25MHZ	1145
3DES	48 级全流水结构的 3DES 加解密算法	156.25MHZ	2530

该拟态异构系统，在网络负载较低及明文的情况下，关闭 FPGA 部分模块，直接由 TOE128 将数据包经 PCIE 传输给 CPU，以降低系统功耗。经测试分析，约可降低 FPGA 总功耗的 3%。在网络负载较高的情况下，启动 FPGA 数据包解析和 3DES 加解密模块，有效降低 CPU 负载，使其有更多的资源处理数据入库，提高系统整体性能。单独使用 CPU 和 CPU+FGPA 的结果对比，如下表所示。

表 6 CPU 和 CPU+FPGA 商品交易性能对比

计算部件	交易量（笔/s）	功耗（W）	能效比（笔/s/W）
CPU	11710	114	102.7
CPU+FPGA	32895	86	382.5

显然，CPU+FPGA 组成的拟态异构系统能效比更高。由于将数据包传输、解析及 3DES 加解密等关键任务移植到了 FPGA 上，在减少数据包响应时延的基础上，利用 FPGA 的并行性和可重构性，提高了处理能力。同时，降低了 CPU 利用率，从 90%左右降至 30%左右，从而使 CPU 拥有更多的空闲资源去处理数据库，提高了每秒交易量。其次，FPGA 属于低功耗器件，而 CPU 随着利用率的下降，其功耗也有明显降低。最后，CPU 和 FPGA 二者各司其职，协同工作，并根据数据包是否加密，变换通路，进一步提高了整个系统的能效比，在性能和功耗之间取得了平衡。

4　结束语

本文提出的基于拟态计算的大数据高效能平台设计方法，通过分析大数据应用的算粒特征，建立计算任务与体系结构匹配矩阵，在应用需求与计算资源间决策出最优的匹配，重构出高效的结构，并以 DVFS 技术降低非关键任务的功耗，以数据布局优化关键任务的功耗，提高系统能效比。实验结果表明该设计方法，突破了一般计算系统性能、效能和灵活性不可兼顾的瓶颈，在充分挖掘计算节点处理能力的同时，降低了系统功耗，使整体能效比达到最优。

但是，对基于拟态计算的大数据高效能平台的研究仍处于初级阶段，未来仍有许多工作需要进一步研究和解决，例如：如何提炼多种不同大数据应用的算粒共性特征，如何实现拟态系统各异构部件间动态调度协作，及如何高效重构映射生成各种大数据算法，即构建超混合可重构计算阵列 HRCA(Hybrid Reconfigurable Computing Array)，进而建立可变的体系结构，满大数据计算的灵活性。

参考文献：

[1] 李国杰，程学旗. 大数据研究：未来科技及经济社会发展的重大战略领域——大数据的研究现状与科学思考 [J]. 中国科学院院刊, 2012, 27 (6): 5-15. (Li Guojie, Cheng Xueqi. Research status and scientific thinking of big data [J]. Bulletin of Chinese Academy of Sciences, 2012, 27 (6): 5-15.)

[2] Dollas A. Big data processing with fpga supercomputers: opportunities and challenges [C]// Proc of IEEE Computer Society Symposium on VLSI. 2014: 474-479.

[3] Nunna K C, Mehdipour F, Trouvé A, *et al.* A survey on big data processing infrastructure: evolving role of FPGA [J]. International Journal of Big Data Intelligence, 2015, 2 (3): 145-156.

[4] Saecker M, Markl V. Big data analytics on modern hardware architectures: a technology survey [C]// Lecture Notes in Business Information Processing. 2013: 125-149.

[5] Neshatpour K, Malik M, Ghodrat M A, *et al.* Energy-efficient acceleration of big data analytics applications using fpgas [C]// Proc of IEEE International Conference on Big Data. 2015: 115-123.

[6] Wang Chao, Li Xi, Chen Peng, *et al.* Heterogeneous Cloud Framework for Big Data Genome Sequencing [J]. IEEE//ACM Trans on Computational Biology & Bioinformatics, 2015, 12 (1): 166-178.

[7] 李鑫，杨学军，徐新海. 分布式流体系结构及其编程模型与资源管理 [J]. 国防科技大学学报, 2015, 37 (6): 110-115. (Li Xin, Yang Xuejun, Xu Xinhai. Programming model and resource management of distributed stream architecture [J]. Journal of National University of Defense Technology, 2015, 37 (6): 110-115.)

[8] 罗建平，谢梦瑶，王华锋. 大数据负载的体系结构特征分析 [J]. 计算机科学, 2015, 42 (11): 48-52. (Luo Jianping, Xie Mengyao. Wang Huafeng. Analysis of architecture characteristics of big data workloads [J]. Computer Science, 2015, 42 (11): 48-52.)

[9] 宫夏屹，李伯虎，柴旭东，等. 大数据平台技术综述 [J]. 系统仿真学报, 2014, 26 (3): 489-496. (Gong Xiayi, Li Bohu, Chai Xudong, *et al.* Survey on big data platform technology [J]. Journal of System Simulation, 2014, 26 (3): 489-496.)

[10] 张东，亓开元，吴楠，等. 云海大数据一体机体系结构和关键技术 [J]. 计算机研究与发展, 2016, 53 (2): 374-389. (Zhang Dong, Qi Kaiyuan, Wu nan, *et al.* Architecture and key technologies of in-cloud smart data appliance [J]. Journal of Computer Research and Development, 2016, 53 (2):

374-389.)

[11] 邬江兴. 拟态计算与拟态安全防御的原意和愿景 [J]. 电信科学, 2014, 30 (7): 1-7. (Wu Jiangxing. Meaning and vision of mimic computing and mimic security defense [J]. Telecommunications Science, 2014, 30 (7): 1-7.)

[12] 沈来信, 王伟. 基于算粒感知的可重构体系结构 [J]. 计算机工程, 2013, 39 (9): 114-118. (Shen Laixin, Wang Wei. Reconfigurable architecture based on operator grain perception [J]. Computer Engineering, 2013, 39 (9): 114-118.)

[13] Liu HaiMing, Wu Qiang. A novel strategy of area cost estimation for custom instruction based on FPGA architecture [C]// Advanced Materials Research. Trans Tech Publications. 2014, 981: 94-98.

[14] Liang Guoqiang, Ma Yuchun, Zhao Kang, *et al.* Efficient custom instruction generation based on characterizing of basic blocks [C]// Proc of IEEE International Conference on Computer Supported Cooperative Work in Design. 2013: 98-103.

[15] 郝水侠, 曾国荪, 谭一鸣. 计算任务与体系结构匹配的异构计算可扩展性分析 [J]. 电子学报, 2010, 38 (11): 2585-2589. (Hao Shuixia, Zeng Guosun, Tan Yiming. Scalability analysis of heterogeneous computing based on computation task and architecture to match [J]. Acta Electronica Sinica, 2010, 38 (11): 2585-2589.)

[16] 梁晓峣. 大数据下处理器体系结构探讨 [J]. 中国计算机学会通讯, 2014, 10 (4): 13-17. (Liang Xiaoyao. Research on processor architecture under big data [J]. Communications of the CCF, 2014, 10 (4): 13-17.)

[17] Pflueger J. Understanding data center energy intensity [R/OL]. (2010) [2017-03-03]. http://www.dell.com/downloads/global/products/pedge/en/Dell-Understanding-Data-Center-Energy-Intensity08122010. pdf.

[18] 林一松, 杨学军, 唐滔, 等. 一种基于关键路径分析的 CPU-GPU 异构系统综合能耗优化方法 [J]. 计算机学报, 2012, 35 (1): 123-133. (Lin Yisong, Yang Xuejun, Tang Tao, *et al.* An integrated energy optimization approach for CPU-GPU heterogeneous systems based on critical path analysis [J]. Chinese Journal of Computers, 2012, 35 (1): 123-133.)

[19] Degalahal V, Tuan T. Methodology for high level estimation of FPGA power consumption [C]// Proc of Design Automation Conference. 2005: 657-660.

[20] 宋杰, 王智, 李甜甜, 等. 一种优化 MapReduce 系统能耗的数据布局算法 [J]. 软件学报, 2015, 26 (8): 2091-2110. (Song Jie, Wang Zhi, Li Tiantian, *et al.* Energy consumption optimization data placement algorithm for MapReduce system [J]. Journal of Software, 2015, 26 (8): 2091-2110.)